Amendment to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claim 1 (currently amended): A scheduling method for a multi-level class hierarchy wherein classes are represented as nodes in a unified tree, said method comprising:

creating a new non-priority node and one or more new priority nodes in said unified tree;

selecting non-priority nodes of said unified tree to establish a non-priority sub-tree, wherein said selected non-priority nodes are children of said new non-priority node in said non-priority sub-tree, as viewed in said multi-level class hierarchy;

selecting priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels, wherein said selected priority nodes are children of said one or more new priority nodes in said one or more priority sub-trees, as viewed in said multi-level class hierarchy;

if and only if queues of nodes of said priority sub-trees are empty, applying a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

if any of said one or more priority sub-trees are non-empty, selecting a priority packet from said one or more priority sub-trees for transmission.

Claim 2 (original): The scheduling method of claim 1 wherein selecting a packet from said one or more priority sub-trees for transmission comprises:

selecting a highest priority non-empty sub-tree from said one or more priority sub-trees; and

applying a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

Claim 3 (original): The scheduling method of claim 2 further comprising:

updating scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

Claim 4 (currently amended): The scheduling method of claim 3 wherein updating comprises:

identifying a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said ~~selected~~ priority sub-tree associated with said ~~transmitted~~ priority packet selected for transmission; and

updating a scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

Claim 5 (original): The scheduling method of claim 4 wherein identifying comprises:

employing a pointer to said identified node.

Claim 6 (original): The scheduling method of claim 4 wherein updating said scheduling state comprises:

adding a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

Claim 7 (currently amended): A ~~computer program product~~ computer-readable medium storing computer-executable instructions for scheduling a multi-level class hierarchy wherein classes are represented as nodes in a unified tree, said ~~product~~ instructions comprising:

code that creates a new non-priority node and one or more new priority nodes in said unified tree

code that causes selection of non-priority nodes of said unified tree to establish a non-priority sub-tree, wherein said selected non-priority nodes are children of said new non-priority node in said non-priority sub-tree, as viewed in said multi-level class hierarchy;

code that causes selection of priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels, wherein said selected priority nodes are children of said one or more new priority nodes in said one or more priority sub-tress, as viewed in said multi-level class hierarchy;

code that, if and only if queues of nodes of said priority sub-trees are empty, causes application of a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission;

code that, if any of said one or more priority sub-trees are non-empty, causes selection of a priority packet from said one or more priority sub-trees for transmission; and

a computer-readable medium that stores the codes.

Claim 8 (currently amended): The computer-readable medium ~~product~~ of claim 7 wherein said code that causes selection of a packet from said one or more priority sub-trees for transmission comprises:

code that causes selection of a highest priority non-empty sub-tree from said one or more priority sub-trees; and

code that causes application of a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

Claim 9 (currently amended): The computer-readable medium ~~product~~ of claim 8 further comprising:

code that causes updating of scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

Claim 10 (currently amended): The computer-readable medium ~~product~~ of claim 9 wherein said code that causes updating comprises:

code that causes identification of a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said selected priority sub-tree associated with said transmitted priority packet; and

code that causes updating of scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

Claim 11 (currently amended): The computer-readable medium ~~product~~ of claim 10 wherein said code that causes identification comprises:

code that causes employment of a pointer to said identified node.

Claim 12 (currently amended): The computer-readable medium ~~product~~ of claim 10 wherein said code that causes updating of said scheduling state comprises:

code that causes addition of a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

Claim 13 (currently amended): Apparatus for scheduling a multi-level class hierarchy wherein classes are represented as nodes in a unified tree, said apparatus comprising:

a processor; and

a computer-readable medium ~~memory~~ storing computer-executable instructions for execution by said processor, said instructions comprising;

code that creates a new non-priority node and one or more new priority nodes in said unified tree;

code that causes selection of non-priority nodes of said unified tree to establish a non-priority sub-tree, wherein said selected non-priority nodes are children of said new non-priority node in said non-priority sub-tree, as viewed in said multi-level class hierarchy;

code that causes selection of priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels, wherein said selected priority nodes are children of said one or more new priority nodes in said one or more priority sub-tress, as viewed in said multi-level class hierarchy;

code that, if and only if queues of nodes of said priority sub-trees are empty, causes application of a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

code that, if any of said one or more priority sub-trees are non-empty, causes selection of a priority packet from said one or more priority sub-trees for transmission.

Claim 14 (original): The apparatus of claim 13 wherein said code that causes selection of a packet from said one or more priority sub-trees for transmission comprises:

code that causes selection of a highest priority non-empty sub-tree from said one or more priority sub-trees; and

code that causes application of a second scheduling algorithm to said highest priority non-empty sub-tree to select a priority packet for transmission.

Claim 15 (original): The apparatus of claim 14 wherein said instructions further comprise:

code that causes updating of scheduling state within said non-priority sub-tree to reflect transmission of said priority packet.

Claim 16 (currently amended): The apparatus of claim 15 wherein said code that causes updating comprises:

code that causes identification of a node within said non-priority sub-tree that has a parent relationship, as viewed in said multi-level class hierarchy, to a node within said ~~selected~~ priority sub-tree associated with said ~~transmitted~~ priority packet selected for transmission; and

code that causes updating of scheduling state of said identified node and ancestor nodes of said identified node within said non-priority sub-tree.

Claim 17 (original): The product of claim 16 wherein said code that causes identification comprises:

code that causes employment of a pointer to said identified node.

Claim 18 (original): The apparatus of claim 16 wherein said code that causes updating of said scheduling state comprises:

code that causes addition of a length of said selected priority packet to a length of a next transmitted packet associated with said identified node to be used in making further scheduling decisions within said non-priority sub-tree.

Claim 19 (currently amended): Apparatus for scheduling a multi-level class hierarchy wherein classes are represented as nodes in a unified tree, said apparatus comprising:

means for creating a new non-priority node and one or more new priority nodes in said unified tree;

means for selecting non-priority nodes of said unified tree to establish a non-priority sub-tree, wherein said selected non-priority nodes are children of said new non-priority node in said non-priority sub-tree, as viewed in said multi-level class hierarchy;

means for selecting priority nodes of said unified tree to establish one or more priority sub-trees corresponding to one or more priority levels, wherein said selected priority nodes are children of said one or more new priority nodes in said one or more priority sub-trees, as viewed in said multi-level class hierarchy;

means for, if and only if queues of nodes of said priority sub-trees are empty, applying a first scheduling algorithm to said non-priority sub-tree to select a packet for transmission; and

means for, if any of said one or more priority sub-trees are non-empty, selecting a priority packet from said one or more priority sub-trees for transmission.